

12º Congresso de Inovação, Ciência e Tecnologia do IFSP - 2021

KOOK - UM APLICATIVO MÓVEL PARA OTIMIZAÇÃO E ORGANIZAÇÃO DE RESTAURANTES USANDO FLUTTER

RENAN VOLPE, ANA PAULA A. C. SHIGUEMORI

1 Graduando em Tecnologia de Análise e Desenvolvimento de Sistemas, PIVICT, IFSP, Câmpus Jacareí, renan.volpe@aluno.ifsp.edu.br

2 Doutora em Computação Aplicada, Professora do Curso de Tecnologia em Análises e Desenvolvimento de Sistemas, Campus Jacareí, anapaula.acs@ifsp.edu.br

Área de conhecimento (Tabela CNPq): 1.03.03.04-7 Análise e desenvolvimento de sistemas.

RESUMO: Este artigo apresenta a implementação de um aplicativo de celular para auxiliar a organização, especificamente, nos atendimentos de um restaurante, tornando-os mais rápidos e mais organizados. Por meio de uma leitura de um QrCode feita pelo próprio celular do cliente, após baixar a aplicação, é capaz de verificar o cardápio e seus respectivos produtos, orientando-se com a foto e a descrição do mesmo. Com base na sua determinada escolha, o chefe de cozinha visualizará os pedidos e seguirá para o preparo dos pedidos e, assim que estiverem prontos, os garçons serão avisados para entregar o pedido até a mesa do cliente. Somado a isso, o Administrador poderá realizar todas as funcionalidades dos seus funcionários além da criação, deleção e atualização de todas as categorias e seus respectivos produtos.

PALAVRA-CHAVE: Flutter; Firebase; Mobile; QrCode; Aplicativo;

KOOK - a mobile app for optimizing and organizing restaurants using Flutter

ABSTRACT: This article presents the implementation of a mobile application to help the organization, specifically, in the service of a restaurant, making them faster and more organized. By reading a QrCode made by the customer's own cell phone, after downloading the application, he is able to check the menu and its respective products, orienting himself with its photo and description. Based on your choice, the chef will view the orders and proceed to prepare the orders, and once they are ready, the waiters will be notified to deliver the order to the customer's table. Added to this, the Administrator will be able to carry out all the functionalities of its employees in addition to the creation, deletion and updating of all categories and their respective products.

Keywords: Flutter; Firebase; *Mobile*; QrCode; Application

INTRODUÇÃO

Com base em algumas aplicações que foram desenvolvidas para melhorar a gestão de restaurantes (NEDEL, 2020; LUCENA, 2020), especificamente no atendimento ao cliente (CAZADO, 2019) e na melhoria de ordenamento dos estabelecimentos (VALENTINO, et al., 2005) mostraram resultados positivos a um grupo de usuários. Esses aplicativos podem ser desenvolvidos para vários ambientes, como por exemplo, Android, IOs, entre outros. Desta forma, a linguagem de desenvolvimento se torna um fator muito importante, para reduzir o tempo gasto no desenvolvimento. Foi realizado um estudo para a escolha de qual linguagem utilizar, como em Análise do desenvolvimento de aplicativos mobile nativos e multiplataforma (PINHEIRO, 2020) e um Sistema integrado para atendimento em restaurantes (NEDEL, 2020); Aplicação mobile para pedidos em bares e restaurantes; Um Aplicativo Móvel para Assistência à Saúde Domiciliar (SILVA, 2017).

Para isso, esse aplicativo visa melhorar a eficiência de entrega de produto mediante a um pedido de um cliente de modo que todas as informações sejam mediadas pelo aplicativo Kook.

O aplicativo, por sua vez, é implementado pela linguagem de programação mobile Dart e Flutter. Além disso, utiliza-se o banco de dados do Firebase para uma interação em tempo real com as atividades de todos os usuários do sistema.

MATERIAIS E MÉTODOS

O projeto, em suas funcionalidades gerais, identifica a mesa que o cliente se apresenta, com ajuda de um QRCode; disponibiliza todo o cardápio do estabelecimento; realiza pedidos que notificam os funcionários; avisa outros funcionários sobre atualização de informação dos pedidos; média o pagamento da cliente; e fazer manutenção de todo o cardápio e seu respectivos produtos.

Com o intuito de diminuir o retrabalho, ou seja, que não tenha a necessidade de criar um programa para cada uma das plataformas, o Flutter foi escolhido para o desenvolvimento desse projeto, que mesmo sendo híbrido se assemelha às tecnologias nativas existentes. A escolha do banco NoSQL com Firebase decorreu por dois pontos: por ser um banco de dados em tempo real (*realtime database*) que se atualizava em momentos contemporâneos do uso da aplicação; e de sua facilidade e desempenho em aplicações com o flutter.

As interfaces do aplicativo mobile foram criadas pela ferramenta de criação de UI (*User interface*) Mockflow. Nessa ferramenta, as páginas do projeto foram originadas em foco nos 3 atores do sistema: cliente, funcionário e o administrador.

Flutter é um *toolkit* de código aberto para desenvolvimento de aplicativos multiplataforma, incluindo os sistemas Android e iOS, sem ser preciso realizar qualquer alteração de desenvolvimento. Este *framework* de tecnologia mobile híbrida, demonstrou uma capacidade de se tornar uma alternativa relevante no desenvolvimento mobile. Por meio de uma comparação realizada em

aplicativos os quais têm as mesmas funcionalidades, no entanto, desenvolvido em aplicações híbridas e nativas, o flutter mostrou-se competitivo ao mercado. Nele existiram diversos destaques que se assemelham e até ultrapassaram a eficiência de um nativo, entre eles: consumo máximo de memória RAM durante o uso do aplicativo, consumo de CPU ao iniciar o aplicativo, tempo de inicialização do aplicativo e tempo para retomar o aplicativo em segundo plano.(PINHEIRO, 2020).

O Firebase Cloud Firestore é um banco de dados NoSQL hospedado na nuvem, designado para aplicativos móveis. O Firebase, faz com que apenas poucos cliques um sistema esteja pronto para ser implementado e se transformado em um sistema ideal para aplicações de rápidas trocas de informações entre o servidor e o usuário acesso. Além disso, o banco garante a sincronia dos dados em aplicativos clientes em tempo real. Ademais, o Cloud Firestore é possível criar consultas superficiais sem precisar recuperar a coleção inteira ou quaisquer outras ou sublocações existentes. Esse meio fornece eficiência na recuperação dos dados.(CORAZZA, 2018)

RESULTADOS E DISCUSSÃO

Durante a implementação de todo o projeto, foi usado o VsCode para a realizar a compilação e a emulação do celular android chamado “Nexus 5X” da marca Lg cuja tela é de 147 x 72,6 mm. Além disso, o projeto é dividido em 3 atores, administrador, cliente e funcionário e teve como base um cardápio de uma loja de frango frito para melhor exemplificação.

A primeira tela do administrador consiste em botões os quais o administrador pode alterar todo o cardápio, pode alterar todos os funcionários e pode verificar a interação dos pedidos ao cliente.

Ao pressionar “Editar cardápio” será possível , deletar e editar uma categoria já existente, além de adicionar uma nova categoria, escolhendo seu ícone e o seu nome. Ao clicar na categoria, serão mostrados todos os produtos dessa categoria que fará as mesmas funcionalidades mais o uso de imagens importadas da câmera e da galeria do celular.

A Figura 1 abaixo contém amostras do uso dessas funcionalidades já com um cardápio criado exemplificadamente.

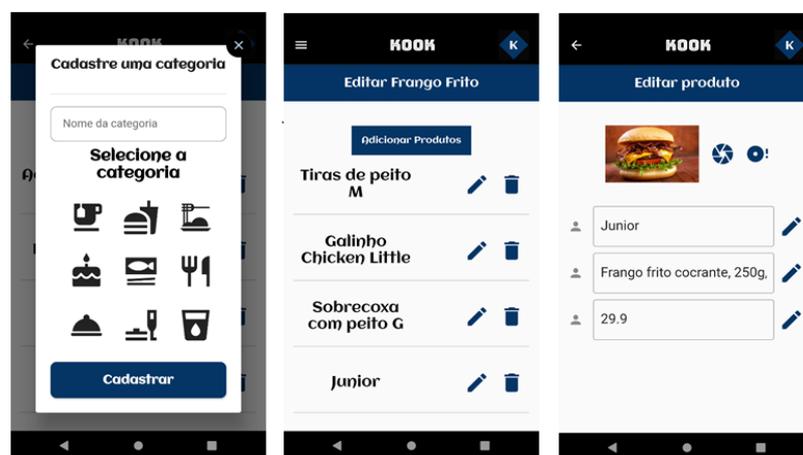


FIGURA 1: Página dos produtos e página de edição de cada produto

Semelhantemente à manutenção das categorias e produtos, no botão “Editar funcionários”, os funcionários são adicionados, alterados e deletados conforme a necessidade do administrador de realizá-la.

A última funcionalidade do administrador é igual à funcionalidade do ator "funcionário". Nessa seção sua função no sistema é de alterar a atual condição do produto. Nessa tela, será possível visualizar a mesa de cada cliente juntamente com o nome do cliente que está sentado na mesa e a observação do pedido. Assim, conforme avançam os pedidos, os 3 atores poderão alterar seu “status” do pedido.

Nessa área, a execução de um código assíncrono, que realiza atualizações constantes, atualiza seus dados rapidamente, aumentando, assim, a fluidez do projeto. A Figura 2 mostra como os pedidos são divididos por cliente junto ao número da mesa e a alteração de um novo *status* de pedido.

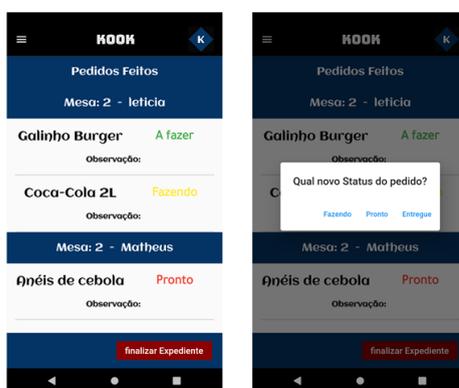


FIGURA 2: Tela dos pedidos e alteração do status de cada pedido

O segmento do cliente começa da mesma maneira que o administrador ou o funcionário em relação a entrada do Login, no entanto, o próprio cliente pode realizar seu cadastro na página primária para realizar as funcionalidades do sistema.

Assim que o cliente fizer o login, ele seguirá escanear, com o próprio aplicativo, a fotografia do QrCode. A classe utilizada para o QrCode foi o “FlutterBarcodeScanner”, uma biblioteca de rápida resposta por ter uma leitura assíncrona.

Com base na leitura, o cliente seguirá na aplicação para escolher o produto que terá preços, fotos e descrições de cada produto. Assim que escolher os produtos que deseja pedir, ele envia para carrinho e confirma, caso tenha certeza de seu pedido. Esse método pode ser realizado a quantidade necessária de vezes até que o sistema se finalize.

Portanto, com a implementação em vigor, o cliente consegue realizar a quantidade necessária de pedidos em um aplicativo instalado no próprio celular e os administrador e seus funcionários conseguem receber os pedidos de forma automatizada e rápida, melhorando assim, a qualidade de um estabelecimento do ramo gastronômico.

A Figura 3 mostra a sequência de ações que o cliente pode realizar no projeto até adicionar o produto ao carrinho.



FIGURA 3: Sequência de ações do cliente no Kook

CONCLUSÃO

Este artigo apresenta um estudo de uma solução mercadológica para estabelecimentos no ramo alimentício. Por meio deste, é possível diminuir o tempo de execução de um pedido em um restaurante e evitar os possíveis erros subjetivos ocorridos durante as interações de garçom e cliente.

Seu objetivo é melhorar a experiência do usuário adequando o estabelecimento às novas tecnologias contemporâneas.

Os resultados preliminares exibidos mostraram que o sistema integra um conjunto de funcionalidades necessárias para que haja integração de códigos assíncrono e uma rápida execução, principalmente relacionados à mudança de *status* dos produtos e as interações entre os usuários. Entretanto, para uma melhor avaliação o sistema será disponibilizado em ambas plataformas, Android e IOs, para testes e validação.

REFERÊNCIAS BIBLIOGRÁFICAS

- CAZADO, A.. **Aplicação Mobile para Pedidos em Bares e Restaurantes**. 2019. Tese (Graduação em Sistema da Informação) - Curso de Sistema da Informação - Universidade Federal de Santa Catarina, Florianópolis, 2019.
- CORAZZA, P. V.. **Um aplicativo multiplataforma desenvolvido com Flutter e NoSQL para o cálculo da probabilidade de apendicite**. 2018. Tese (Graduação em Ciência da Computação) - Curso Bacharel em Ciência da Computação - Universidade Federal do Rio Grande do Sul, Porto Alegre, 2018.
- LUCENA, F. A.. **Controli: aplicativo para gestão e consolidação financeira**. Dissertação (Pós-Graduação em Ciência, Tecnologia e Inovação) - Mestrado em Ciência, Tecnologia e Inovação - Universidade Federal do Rio Grande do Norte, Natal, 2020.
- NEDEL, M. **Sistema Integrado para Atendimento em Restaurantes**. 2020. Tese (Graduação em Sistema da Informação) - Curso de Sistema da Informação - Universidade do Sul de Santa Catarina, Florianópolis, 2020.
- PINHEIRO, J. S.. **Análise do desenvolvimento de aplicativos mobile nativos e multiplataforma**. 2020. Tese (Graduação em Ciência da Computação) - Curso de Ciência da Computação - Universidade Federal de Santa Catarina, Florianópolis, 2020.
- VALENTINO, L.; SCHNEIDER, H.; SCHELL, R. **Aplicações móveis: Arquitetura, projeto e desenvolvimento**. 1. ed. São Paulo: Editora Pearson, 2005